

Teoría de las comunicaciones

Práctica 7: Congestión

Temas

Conceptos generales de congestión, Control de congestión en TCP (RFC5681)

Definiciones

Ventana máxima:

$$MaxWindow = Min(RWND, CWND)$$

Ventana efectiva:

$$EffectiveWindow = MaxWindow - (LastByteSent - LastByteAcked)$$

Ventana de congestión inicial:

$$CWND = IW = 2 * SMSS$$

Tamaño de segmento inicial:

$$SMSS = 2KB$$

Tamaño de Ssthresh inicial:

$$ISsthresh = 64kb$$

Incremento de la ventana de congestión en Slow Start ($CWND < Ssthresh$):

$CWND+ = min(N, SMSS)$ por cada ACK con N la cantidad de bytes reconocidos por el ACK.

Incremento de la ventana de congestión en Congestion Avoidance ($CWND > Ssthresh$):

$CWND+ = SMSS * SMSS / CWND$ por cada ACK o hasta un 1 $SMSS$ por RTT .

Decremento de la ventana y el umbral ante un time-out:

$$Ssthresh = max(FlightSize/2, 2 * SMSS)$$

$$CWND = LW = 1 SMSS$$

Decremento de la ventana y el umbral ante 3 ACKs duplicados (FR/FR):

$$Ssthresh = max(FlightSize/2, 2 * SMSS)$$

$$CWND = Ssthresh + 3$$

Incremento de la ventana de congestión en FR/FR:

$CWND+ = SMSS$ por cada ACK Duplicado.

Salida de FR/FR:

$CWND = Ssthresh$ ante un ACK que reconoce datos nuevos.

Reinicio de la ventana ante un tiempo idle (1 RTO sin transmitir):

$$RW = min(IW, cwnd)$$

Estimación del RTT:

$RTT[i+1] = (1 - alpha) * RTT[i] + alpha * SRTT$ siendo $SRTT$ el $i - esimo$ RTT medido

Ajuste del RTO:

$$RTO[i] = 2 * RTT[i]$$

Ejercicio 1

Dado un protocolo de nivel de red basado en forwarding de datagramas (i.e.: IP):

- a. ¿Se puede realizar control de congestión utilizando la técnica de prealocación de buffers?
- b. ¿Los protocolos de ruteo pueden por sí solos evitar la congestión en una red?
- c. En el caso de disponer de buffers con memoria infinita en los routers. ¿Se puede evitar la congestión en una red?

Ejercicio 2

El control de congestion de TCP usa la CWND como una estimación de la congestión presente en la red.

- a. ¿En qué casos no es necesario un control de congestión?
- b. Mencione dos situaciones en las que la CWND no sea representativa de la congestión de la red. ¿Qué medidas deberían tomarse ante estas situaciones?

Ejercicio 3

CWND y Ssthresh son las dos variables que deben regularse en el control de congestión de TCP.

- a. ¿Qué valores deberían tener cuando la conexión recién comienza?
- b. ¿Cómo se modifican ante un time out?
- c. Slow Start se usa cuando no se conoce el estado de la congestión de la red. ¿Por qué no esta bien empezar con una ventana arbitrariamente grande?
- d. ¿Cuánto debe incrementarse la ventana de congestión por RTT durante Congestion Avoidance?

Ejercicio 4

En una conexión TCP recién establecida se envían estos segmentos en el siguiente orden y con los siguientes tamaños:

100 bytes ; 150 bytes ; 200 bytes ; 300 bytes

Todos los segmentos llegan al receptor sin errores pero el segundo segmento llega último.

- a. Indicar hasta qué byte reconoce el receptor en cada ACK.
- b. Indicar cuál es el valor del Ssthresh, posterior a la recepción del ACK que indica que los cuatro segmentos llegaron a destino.
- c. Indicar cuál es el valor del CWND, posterior a la recepción del ACK que indica que los cuatro segmentos llegaron a destino.

Ejercicio 5

Suponga que, para una conexión, cuatro ACKs fueron devueltos con los siguientes RTTs: [80ms, 20ms, 100ms, 90ms].

- Calcule el RTO luego de la llegada de cada ACK usando $\alpha = 1,0$.
- Repita el cálculo para $\alpha \in [0,8, 0,6, 0,4, 0,2]$.
- ¿Se produce algún time-out en alguno de los escenarios anteriores?

Tomar el RTT inicial como la primera medición

Ejercicio 6

Considere el efecto de usar Slow Start en una conexión TCP recién establecida ($IW = 2 * SMSS$, $SSTHRESH = 64KB$), que tiene un RTT de 10 mseg y sin congestión ni errores presentes en la red. La $RWND$ es de 24KB y el $SMSS$ es de 2KB. ¿Cuánto tiempo transcurre antes de que pueda ser enviada la primera ventana de recepción llena? (*Asumir que el Ttx de una ventana es una componente despreciable del Delay total de la conexión*)

Ejercicio 7

Según el RFC 5681,

- Durante slow start, la ventana de congestión DEBERÍA incrementarse usando $CWND+ = \min(N, SMSS)$. Sin embargo, según el RFC 2581, se usaba la fórmula $CWND+ = SMSS$. ¿A que se debió este cambio?
- En una conexión TCP, el receptor PUEDE demorar los ACKs hasta haber recibido 2 SMSS del emisor o hasta 500ms, lo que suceda primero. ¿Qué impacto tienen estas demoras en el emisor?

Ejercicio 8

Grafique $CWND$ en función del tiempo para una conexión TCP que pierde paquetes cada 1 seg para los siguientes casos detallando los valores de $SSTHRESH$:

- Sin Fast Retransmit / Fast Recovery
- Con Fast Retransmit / Fast Recovery

Ejercicio 9

Una conexión TCP tiene que enviar 1GB de datos. Suponiendo que no se producen errores en la transmisión de los datos y que las únicas pérdidas de paquetes son por congestión. ¿Calcular cuánto tiempo tarda la conexión en enviar todos los datos para los siguientes casos?

- Congestión con 32KB por RTT o mas
- Congestión con 16KB por RTT o mas
- Sin congestión (i.e.: una conexión establecida entre hosts de la misma LAN)

Asumir: si una ráfaga causa congestión se pierden todos los segmentos de la misma antes de llegar al receptor y que la $RWND$ vale siempre 64KB

Ejercicios de Parcial

Ejercicio 10

En una conexión recién establecida con $RTT=200ms$, el host receptor siempre anuncia una *AdvertisedWindow* de 16KB. La red está cargada al punto que si una ráfaga fuera de 16KB o mas, se perderían todos los segmentos de la misma.

- ¿Cuánto vale la $CWND$ luego de enviar un archivo de 40KB?
- 3 segundos después del envío del archivo, se envía otro archivo de 30KB ¿Cuánto tiempo tarda?

Ejercicio 11

Una conexión con $RTT=400ms$, $SSTHRESH=24KB$ y $CWND=64KB$, se encuentra idle durante 3 segundos y le llegan para transmitir 60KB de datos. Asumiendo que se congestiona la red sobre la que transmite cuando una ráfaga supera los 16KB perdiendo todos los segmentos de la ráfaga.

- a. ¿Cuánto tiempo le toma a la conexión el envío de los datos?
- b. ¿Cuál es la desventaja de usar Fast Retransmit/Fast Recovery en una conexión que pasa por una red que desordena muchos paquetes?

Ejercicio 12

Una conexión ya establecida con $RTT=200\text{ms}$, se encuentra Idle durante 3 segundos hasta que necesita enviar 50KB de datos. Tiene $SSTHRESH=8\text{KB}$, $CWND=10\text{KB}$ y el receptor siempre anuncia 24KB de *Advertised Window*. No hay pérdida de paquetes por congestión.

- Si se define la velocidad de transmisión de datos (Throughput) de una conexión como el tamaño de la ventana efectiva dividido el RTT (ej: $Throughput = MaxWindow/RTT$) ¿Cuál es la máxima velocidad alcanzada por la conexión?
- ¿Cómo se clasifica el control de congestión de TCP (**RFC 5681**) dentro de la clasificación MIMD/AIMD/MIAD/AIAD? (*M: Multiplicative, A: Additive, I: Increase, D: Decrease*)

Ejercicio 13

En un host emisor, una conexión TCP recién establecida con $RTT=100\text{ms}$, necesita enviar 60KB. La conexión pasa a través de una red que está lo suficientemente cargada como para que cada vez que una ráfaga de segmentos fuera de 20KB o mas en un RTT, todos ellos se pierden.

- Si el receptor anuncia, en sus ACKs, una *Advertised Window* de 32KB hasta haber recibido un total de 14KB y, a partir de ahí, sus ACKs anuncian 16KB ¿Cuánto tiempo tarda la conexión en enviar todos los datos?
- Si el receptor hubiera anunciado siempre 32KB, el envío de los datos, ¿hubiera tardado más o menos tiempo?

Bibliografía

Computer Networks: A systems approach. 3ra Edición. *Peterson & Davie*. Capítulo 6: Congestion Control and Resource Allocation (sección 6.3).

RFC 5681: TCP Congestion Control.